U.S. Patent Application Serial No. 09/599,806, entitled "Methods and Systems of Providing Information to Computer Users", bearing attorney docket number MS1-557us, and filed on the same date as this patent application;

- U.S. Patent Application Serial No. 09/599,299, entitled "Methods, Systems, Architectures and Data Structures For Delivering Software via a Network", bearing attorney docket number MS1-559us, and filed on the same date as this patent application;
- U.S. Patent Application Serial No. 09/599,048, entitled "Network-based Software Extensions", bearing attorney docket number MS1-563us, and filed on the same date as this patent application;
- U.S. Patent Application Serial No. 09/599,813, entitled "Authoring Arbitrary XML Documents Using DHTML and XSLT", bearing attorney docket number MS1-583us, and filed on the same date as this patent application;
- U.S. Patent Application Serial No. 09/599,086, entitled "Task Sensitive Methods And Systems For Displaying Command Sets", bearing attorney docket number MS1-562us, and filed on the same date as this patent application.--

In the Claims

Claims 1, 9, 12, 18, 23, 30, 31 and 37 are amended.

Claims 1-42 are pending and are listed below as follows:

1. (Amended) A software architecture embodied on a computer-readable medium, the architecture comprising:

multiple attachment points collectively arranged to filter data associated with files that describe software extensions; and

multiple extension managers associated with the multiple attachment points and with respective feature types that can be added to a software platform by software extensions, the extension managers being configured to receive data from

the multiple attachment points that pertains only to the feature type with which the extension manager is associated.

2. The software architecture of claim 1, wherein the attachment points are defined as predicate chains.

3. The software architecture of claim 1, wherein the attachment points filter XML data.

4. The software architecture of claim 3, wherein each feature type is associated with an XML tag.

5. The software architecture of claim 3, wherein each feature type is associated with an XML tag, at least some of the feature types comprising user-defined feature types.

6. The software architecture of claim 1, wherein each attachment point exposes collections of ordered nodes.

7. The software architecture of claim 1, wherein each attachment point exposes collections of ordered XML nodes.

8. A computer embodying the software architecture of claim 1.

9. (Amended) A software architecture embodied on a computer-readable medium, the architecture comprising:

a hub structure configured to:

receive multiple different files that describe software extensions that can be added to a software platform;

combine the multiple different files into a single exposable list; and

expose the single exposable list to a filter structure that is configured to filter the list.

10. The software architecture of claim 9, wherein the hub structure receives multiple different XML files and exposes a list of XML nodes.

11. A computer embodying the software architecture of claim 9.

12. (Amended) A software architecture embodied on a computer-readable medium, the architecture comprising multiple different attachment points each of which is configured to:

receive XML data that pertains to one or more software extensions that can be added to a software platform;

process the XML data to provide a list of XML nodes; and

expose the list of XML nodes.

13. The software architecture of claim 12, wherein the list of XML nodes is exposed to another attachment point.

14. The software architecture of claim 12, wherein the list of XML nodes can pertain to multiple different feature types that can be added by the one or more software extensions.

15. The software architecture of claim 12, wherein the list of XML nodes can pertain to multiple different features of particular feature types that can be added by the one or more software extensions.

16. The software architecture of claim 12, wherein the list of XML nodes can pertain to one or more of:

multiple different feature types that can be added by the one or more software extensions; and

multiple different features of particular feature types that can be added by the one or more software extensions.

17. A computer embodying the software architecture of claim 12.

18. (Amended) A software architecture embodied on a computer-readable medium, the architecture comprising:

a hub structure configured to:

receive multiple different files that describe software extensions that can be added to a software platform;

combine the multiple different files into a single exposable list; and

expose the single exposable list to a filter structure that is configured to filter the list;

a filter structure comprising multiple attachment points collectively arranged to filter data associated with the list exposed by the hub structure; and

multiple extension managers associated with the multiple attachment points and with respective feature types that can be added to a software platform by software extensions, the extension managers being configured to receive data from the multiple attachment points that pertains only to the feature type with which the extension manager is associated.

19. The software architecture of claim 18, wherein the hub structure receives multiple different XML files and exposes a list of XML nodes.

20. The software architecture of claim 19, wherein the list contains root node tags for all of the XML files.

21. The software architecture of claim 19, wherein the XML files logically describe where a particular extension fits on the software platform.

22. The software architecture of claim 19, wherein the attachment points are defined as predicate chains.

23. (Amended) The software architecture of claim 19, wherein an extension manager is notified whenever an extension comprising a feature type with which it is associated is added or removed from the software platform.

24. The software architecture of claim 19, wherein each feature type is associated with a particular XML tag.

25. A computer embodying the software architecture of claim 18.

26. A method of providing a software extension comprising:

exposing an XML list that contains one or more nodes;

processing the XML list to identify specific nodes that correspond to various feature types that can be added to a software platform; and

notifying an extension manager that is associated with at least one feature type if a node that corresponds to that feature type is identified in the XML list.

27. The method of claim 26, wherein said processing is accomplished by filtering the XML list using multiple attachment points that are defined as predicate chains.

28. The method of claim 27, wherein the individual attachment points receive XML data as an input and expose a list of XML nodes.

29. The method of claim 26, wherein said processing is accomplished by filtering on specific nodes.

30. (Amended) The method of claim 26, wherein said processing is accomplished by exposing various nodes.

31. (Amended) The method of claim 26, wherein said processing is accomplished by filtering on specific nodes and exposing various nodes.

32. One or more computer-readable media having computer-readable instructions thereon which, when executed by a computer, cause the computer to implement the method of claim 26.

33. A method of providing a software extension comprising:

receiving XML data that pertains to a software extension that is to be added to a software platform;

processing the XML data to identify XML nodes; and

exposing an XML list that contains one or more nodes that are identified by said processing.

34. The method of claim 33, wherein said receiving comprises receiving multiple XML files that pertain to different software extensions.

35. The method of claim 34, wherein said processing comprises combining the multiple XML files into a single exposable list.

36. The method of claim 33, wherein said processing comprises processing the XML data with one or more attachment points that are defined as predicate chains that filter the XML data.

37. (Amended) The method of claim 36, wherein at least one of the attachment points exposes a node.

38. The method of claim 36, wherein at least one of the attachment points filters on a node.

39. One or more computer-readable media having computer-readable instructions thereon which, when executed by a computer, cause the computer to implement the method of claim 33.

40. A method of providing a software extension comprising:

receiving multiple different files, each of which being associated with a different software extension and logically describing its associated software extension;

combining the multiple different files in a single list;

exposing portions of the list;

processing exposed portions of the list to identify one or more feature types that are to be added to a software platform; and

notifying an extension manager that is associated with a particular feature type.

41. The method of claim 40, wherein the multiple different files comprise XML files.

42. One or more computer-readable media having computer-readable instructions thereon which, when executed by a computer, cause the computer to implement the method of claim 40.